Carrie Magee, Akshay Verma, Jack McMorrow, Paul Kelly

Using Deep Learning to Bridge Language and Numbers

1. Introduction

Inherent ambiguity serves as the epitome of complexity in human language, and the introduction of math and language, in the context of word problems, further accentuates this intricacy. Nevertheless, our group aimed to confront this challenge by leveraging the power of transformers to unravel the intricacies woven into the phenomena of mathematical linguistics.

Math word problems, comprising short natural language narratives, present exercises that often involve real-world scenarios. These scenarios require individuals to apply their skills to interpret and solve problems, requiring not only a grasp of basic math but also an understanding of context, sentence structure, and word dependencies. The process of solving math problems entails understanding the given problem, prioritizing relevant information, and converting the written language problem into solvable mathematical expressions. Considering the intricate steps involved in solving word problems, it is evident that they pose a distinctive challenge to computers, primarily due to the necessity for ambiguity resolution and contextual understanding.

In our attempt to solve this problem, the multimodal capacities of transformers emerged as valuable assets. The goal of our project is to solve linguistic challenges with computational solutions, more specifically use the power of deep learning to convert word problems into solvable mathematical equations.

1. Description of dataset

The dataset used is [SVAMP](https://github.com/arkilpatel/SVAMP) (Simple Variation on Arithmetic Math word Problems). The dataset was created after models were found to rely on shallow heuristics in benchmark word math problem datasets like [ASDiv-A](https://github.com/chaochun/nlu-asdiv-datase) and [MAWPS](https://github.com/sroy9/mawps). The SVAMP dataset contains 1000 word problems, we will be using this data set for evaluation. The ASDiv-A and MAWPS dataset contains 1921 and 1217 word problems respectively.

1. Description of the NLP model and what kind of algorithm you use. Provide some background information on the development of the algorithm and include necessary equations and figures.
2. Experimental setup. Describe how you are going to use the data to train and test the model. Explain how you will implement the model in the chosen framework and how you will judge the performance.
3. What kind of hyper-parameters did you search on? (e.g., learning rate)? How will you detect/prevent overfitting and extrapolation?
4. Results. Describe the results of your experiments, using figures and tables wherever possible. Include all results (including all figures and tables) in the main body of the report, not in appendices. Provide an explanation of each figure and table that you include. Your discussions in this section will be the most important part of the report.
5. Summary and conclusions. Summarize the results you obtained, explain what you have learned, and suggest improvements that could be made in the future.
6. References. In addition to references used for background information or for the written portion, you should provide the links to the websites or github repos you borrowed code from.
   1. <https://github.com/arkilpatel/SVAMP>
   2. https://huggingface.co/t5-small